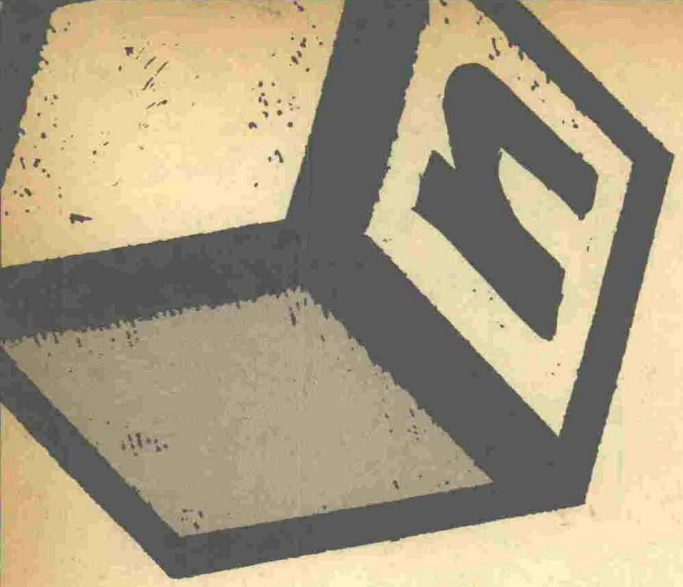
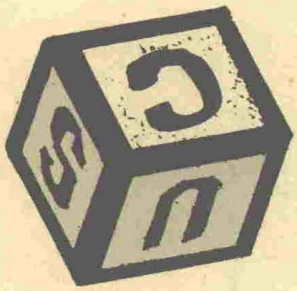
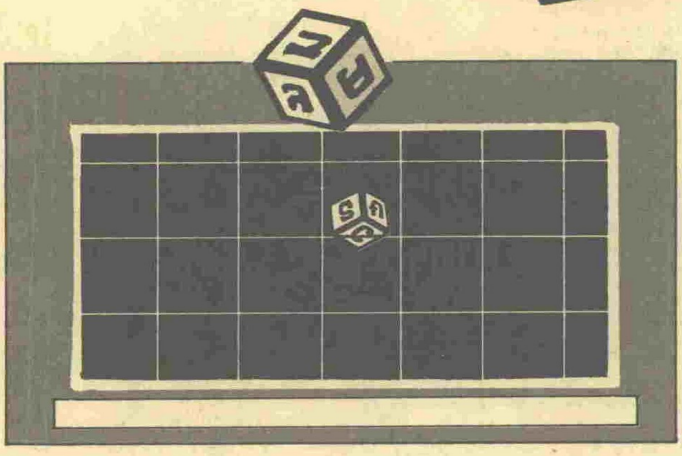






การสร้าง ภาษาไทยกราฟิก

โปรแกรมเขียนภาษาไทยกราฟิก 25 บรรทัดด้วยแอสเซมบลีจะดีกว่าซอฟต์แวร์ที่เขียนสำหรับการใช้การ์ดภาษาไทยที่ใช้กับฮาร์ดแวร์นั้น ๆ เฉพาะของบริษัทแต่ละบริษัท

สุวิทย์ มณฑลผลิน

การแสดงผลภาษาไทยบนจอภาพจะสามารถแสดงได้ทั้งในโหมดตัวอักษร (Text Mode) และในโหมดกราฟิก ซึ่งแต่ละโหมดจะมีข้อดี และข้อเสียแตกต่างกันไป

ในโหมดเท็กซ์นั้น รูปแบบของตัวอักษร (Font) จะถูกเก็บไว้ในคาแรกเตอร์เยนเนอเรเตอร์ ซึ่งอยู่บนการ์ด CRT Controller โดยมีไอซีเบอร์ 6845 เป็นตัวควบคุมจัดการเกี่ยวกับการแสดงผลทั้งหมด ซึ่งปกติจะสามารถ แสดงได้ 25 บรรทัด 80 คอลัมน์ การแสดงผลในลักษณะนี้จะได้ผลที่รวดเร็วเนื่องจาก CPU ไม่ต้องเสียเวลามาจัดการเกี่ยวกับจอภาพ ปล่อยให้เป็นหน้าที่ของ CRT Controller โดย CPU เพียงแต่ส่งรหัสของตัวอักษร และแอตทริบิวต์ที่ต้องการไปไว้ในส่วนของหน่วยความจำของจอภาพเท่านั้น

แต่การแสดงผลของภาษาไทยนั้นจะแตกต่างจากการแสดงผลของภาษาอังกฤษ เพราะภาษาไทยจะประกอบด้วย พยัญชนะ สระ และวรรณยุกต์ การแสดงผลจึงเป็นการแสดงในแบบ 3 ระดับ คือจะแสดงในลักษณะของ สระบนและวรรณยุกต์จะอยู่ในบรรทัดบน พยัญชนะและสระกลางจะอยู่บรรทัดต่อมา และสระล่างจะใช้อีก 1 บรรทัด ดังนั้นการแสดงผลภาษาไทย 1 บรรทัดจะเท่ากับการแสดงผลของภาษาอังกฤษถึง 3 บรรทัด ในโหมดเท็กซ์ ซึ่งมี 25 บรรทัด จึงสามารถแสดงภาษาไทยได้เท่ากับ 25/3 คือ 8 บรรทัดและมีเหลืออีก 1 บรรทัดของภาษาอังกฤษ เราอาจจะแสดงผลได้มากกว่า 8 บรรทัด โดยการนำสระล่างของบรรทัดบน มาไว้ปนกับสระบนของบรรทัดล่าง แต่จะเกิดปัญหาในกรณีที่สระตรงกันการแสดงผลที่หลังจะไปทับครั้งแรก จึงต้องใช้วิธีเลี่ยงสระตัวหนึ่งตัวใดออกไป ทำให้อ่านได้ยาก และไม่สวยงาม แต่จะใช้ 2 บรรทัดของภาษาอังกฤษ แทน 1 บรรทัดของภาษาไทย จึงสามารถแสดงภาษาไทยได้ 12 บรรทัด

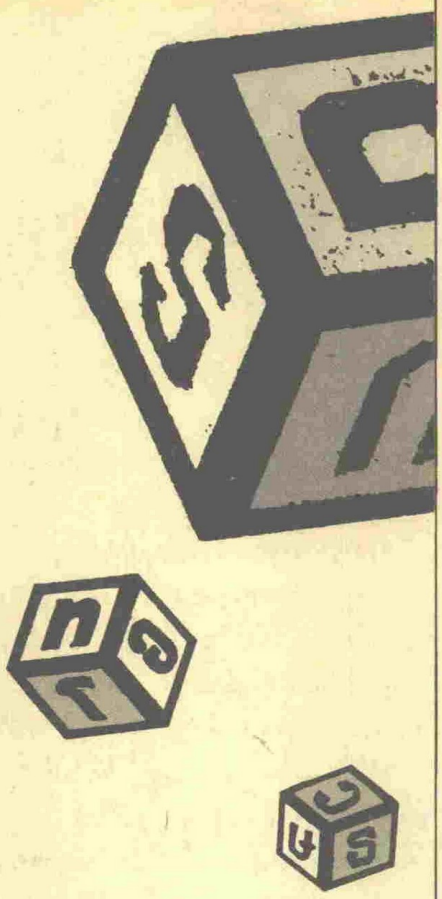
การแสดงผลภาษาไทยในเท็กซ์-โหมด ตามที่กล่าวมานี้จะกระทำโดยนำรูปแบบของตัวอักษรที่จะแสดงเก็บไว้ในคาแรกเตอร์เยนเนอเรเตอร์ ซึ่งปกติจะเก็บรหัส ASCII ตั้งแต่ 0-127 เป็นรหัส ASCII มาตรฐาน และรหัส 128-255 ใช้เก็บรหัสขยายกราฟิก ซึ่ง

ไม่ค่อยจะได้ใช้ เราจึงนำรูปแบบของรหัสภาษาไทยมาเก็บไว้ในรหัสช่วงหลังนี้ แต่บางครั้งก็ยังมีความต้องการใช้รหัสขยายกราฟิกอยู่ ดังนั้นจึงมีการสร้างคาแรกเตอร์เยนเนอเรเตอร์ ขึ้น 2 ชุด แล้วใช้ฮาร์ดแวร์หรือซอฟต์แวร์-สวิตช์ เป็นตัวเลือกว่าจะใช้รหัสในชุดใด

การแสดงภาษาไทยในโหมดเท็กซ์นี้จะสามารถแสดงได้เพียง 8 บรรทัด หรือ 12 บรรทัดเท่านั้น อีกทั้งยังไม่สวยงาม เพราะสระบนหรือสระล่างจะอยู่ห่างจากพยัญชนะมาก การอ่านก็ยากและยังต้องนำเครื่องที่ใช้ไปติดตั้งคาแรกเตอร์เยนเนอเรเตอร์ภาษาไทยอีกด้วย ต่อมามีการติดตั้งการ์ดภาษาไทย 25 บรรทัด ซึ่งสามารถที่จะจัดระดับของสระได้เอง แต่ตัวหนังสือที่แสดงจะมีลักษณะตัวเล็กไม่สวยงาม และมีการกะพริบหรือสั่นของจอภาพเกิดขึ้นด้วย ซึ่งเมื่อใช้ไปนาน ๆ จะปวดตา และส่งผลต่อสายตาในระยะยาว นอกเหนือจากนั้นเนื่องจากมีการผลิตการ์ดภาษาไทยกันหลายบริษัทซึ่งแต่ละบริษัทก็จะมีมาตรฐานของตนเอง จึงเกิดปัญหาตามมาคือโปรแกรมหรือซอฟต์แวร์ที่เขียนสำหรับใช้กับการ์ดหนึ่ง ๆ จะไม่สามารถนำไปใช้กับการ์ดอื่น ๆ ได้ ซึ่งเรียกว่าโปรแกรมนั้น ๆ ขึ้นอยู่กับฮาร์ดแวร์ซึ่งไม่สะดวกต่อการใช้งานเป็นอย่างมาก

ภาษาไทยกราฟิกดีอย่างไร

การแสดงผลภาษาไทยในโหมดกราฟิกนี้กระทำโดยการสร้างรูปแบบของตัวอักษรที่จะแสดงขึ้นมาแล้วนำไปเก็บไว้ในหน่วยความจำ เมื่อต้องการที่จะแสดงผล CPU จะกระทำตามโปรแกรมจัดการย้ายรูปแบบออกปรากฏบนจอภาพ พร้อมกับทำการจัดระดับในการแสดงให้ด้วย



การแสดงผลกราฟิกในจอโมโนโครม โดยทั่ว ๆ ไปจะยึดหลักการแสดงผลของการ์ดเฮอร์คิวลิสซึ่งดูเหมือนจะกลายเป็นมาตรฐานโดยปริยายแล้ว ดังนั้นไม่ว่าจะเป็นการ์ด CRT Controller ของบริษัทใด ๆ ก็ตาม การแสดงผลในโหมดกราฟิกจะเหมือนกันหมด ดังนั้นการแสดงผลภาษาไทยในโหมดกราฟิกไม่ขึ้นอยู่กับการ์ดที่แสดง จึงไม่ก่อให้เกิดปัญหาซอฟต์แวร์ผูกพันกับฮาร์ดแวร์

รูปแบบของตัวอักษรที่จะแสดงออกบนจอภาพ ในโหมดกราฟิกเราสามารถแสดงได้ในหลายรูปแบบ เช่น ขีดเส้นใต้ หรือตัวขยาย ซึ่งจะอยู่กับตัวโปรแกรมจัดการหน้าจอนี้ การแสดงผลแบบนี้ในเท็กซ์โหมดไม่มีทางทำได้ นอกจากนี้ยังสามารถนำภาพกราฟิกมาแสดงร่วมกับตัวอักษรเหล่านี้ได้ด้วย

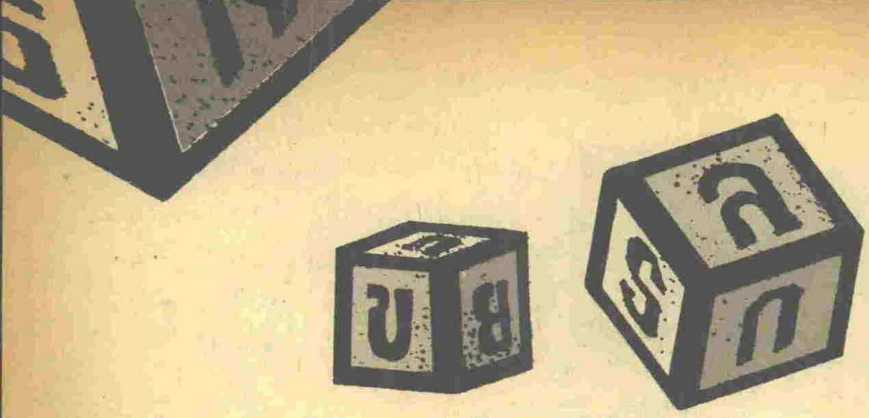
ส่วนข้อเสียของการแสดงในโหมดกราฟิกก็คือ CPU จะต้องเสียเวลาส่วนหนึ่งมาจัดการในการนำฟอนต์ออกแสดงบนจอภาพ โปรแกรมส่วนที่จัดการจอภาพนี้ควรมีขนาดเล็ก ทำงานได้รวดเร็ว ทำให้เสียเวลาของ CPU น้อยที่สุด ดังนั้นโปรแกรมส่วนที่จัดการหน้าจอนี้จึงควรเขียนด้วยภาษาแอสเซมบลี้

หลักการของโปรแกรม

การแสดงกราฟิกของการ์ดเฮอร์คิวลิสจะแบ่งการแสดงออกเป็น 2 หน้า คือ หน้าศูนย์ และหน้าหนึ่ง โดยใช้หน่วยความจำสำหรับแสดงในหน้าศูนย์ที่ B0000-B7FFF และหน้าหนึ่งที่หน่วยความจำ B8000-BFFFF ซึ่งหน่วยความจำในการแสดงผลในหน้าศูนย์จะมีหน่วยความจำที่ซ้ำกับการแสดงผลของโหมดเท็กซ์ ดังนั้นโปรแกรมนี้จึงยึดหลักการแสดงผลในโหมดกราฟิกที่หน้าหนึ่งเป็นหลัก แต่ก็สามารถแสดงได้ทั้ง 2 หน้าซึ่งจะกล่าวต่อไป ส่วนการแสดงผลของการ์ดเฮอร์คิวลิสนั้นสามารถศึกษาได้จากวารสารไมโครคอมพิวเตอร์ ฉบับที่ 29

การ์ดเฮอร์คิวลิสนั้นสามารถแสดงความละเอียดในโหมดกราฟิกขนาด 720 × 348 การแสดงผลภาษาไทยจะคำนึงถึงความสวยงามด้วย ดังนั้นจะใช้ฟอนต์ซึ่งมีขนาด 80 × 20 โดยทางแนวตั้งนี้จะแบ่งสแกนไลน์ทั้ง 20 เส้นออกเป็นวรรณยุกต์และสระบน 8 เส้น พยัญชนะอีก 8 เส้น สระล่างอีก 4 เส้น การแสดงผลจะสามารถแสดงได้ 720/8 = 90 คอลัมน์ และ 348/20 = 17 บรรทัด และเหลืออีก 8 เส้น ซึ่งจะไม่ได้ใช้อยู่ทางตอนล่างของจอภาพ

เพื่อความเป็นอิสระในการติดต่อกับภาษาไทยกราฟิกนี้ให้สามารถเรียกใช้จากภาษาขั้นสูงได้สะดวก ดังนั้นจึงทำเป็นโปรแกรมเรสซิเดนต์ฝังไว้เป็นส่วนของดอส โปรแกรมที่จะติดต่อใช้งานภาษาไทยกราฟิกนี้จะติดต่อผ่านทางอินเตอร์รัพต์หมายเลข 1FH (Graphics Character Extensions) ซึ่งเป็นอิน-



เตอร์รัพต์ชี้ตารางตัวอักษรกราฟิก การใช้งานอินเตอร์รัพต์ 1FH นี้จะมีฟังก์ชันย่อยให้เลือกอีก 9 ฟังก์ชันคือ ฟังก์ชัน 0 ถึงฟังก์ชัน 8 โดยการใช้งานจะใช้รีจิสเตอร์ AH กำหนดว่าเป็นฟังก์ชันใดและค่าต่าง ๆ ที่ฟังก์ชันต้องการจะถูกกำหนดไว้ในรีจิสเตอร์ ซึ่งจะกล่าวถึงรายละเอียดในการใช้งานฟังก์ชันเหล่านี้ต่อไป เมื่อมีการเรียกอินเตอร์รัพต์มา โปรแกรมจะนำค่าในรีจิสเตอร์ AH มาตรวจดูว่าเป็นฟังก์ชันใด และส่งการทำงานไปยังฟังก์ชันนั้น ๆ

รหัสภาษาไทยกราฟิกนี้ใช้รหัสของสำนักงานมาตรฐานอุตสาหกรรม สามารถแสดงตัวอักษรได้หลายรูปแบบตามลักษณะของการเซตแอตทริบิวต์ เนื่องจากแสดงในโหมดกราฟิกดังนั้นจึงสามารถนำภาพกราฟิกมาแสดงร่วมได้ ดังนั้นจึงออกแบบให้โปรแกรมสามารถที่จะนำฟอนต์ของตัวอักษรออกแสดงบนหน้าจอในลักษณะของ X-OR หรือ Overwrite โปรแกรมภาษาไทยกราฟิกนี้ใช้วิธีจัดการหน้าจอเอง โดยคำนวณตำแหน่งบนหน้าจอภาพ ตำแหน่งของตัวอักษร ให้สัมพันธ์กับตำแหน่งของเคอร์เซอร์ และโปรแกรมจะต้องทำการจัดระดับของอักขระด้วยในการพิมพ์ตัวอักษรและสระกลาง เมื่อทำการพิมพ์แล้วจะต้องเลื่อนตำแหน่งของเคอร์เซอร์ไป 1 ตำแหน่ง แต่ถ้าเป็นสระล่างหรือสระบนหรือวรรณยุกต์จะต้องถอยหลังเคอร์เซอร์กลับมา 1 ตำแหน่ง ทำการพิมพ์ในลักษณะ X-OR กับตัวอักษรที่อยู่ก่อน จากนั้นจึงทำการเลื่อนตำแหน่งของเคอร์เซอร์ไปดังเดิม

ฟังก์ชันการแสดงผล

ในโปรแกรมภาษาไทยกราฟิกจะมีฟังก์ชันสำหรับการแสดงผลอยู่หลายฟังก์ชันประกอบด้วย

ฟังก์ชัน 0 เลือกหน้าของการแสดงผล

การแสดงผลของภาษาไทยกราฟิกนี้จะยึดการแสดงในหน้าหนึ่งของการ์ดเฮอร์คิวลิสเป็นหลัก เมื่อเริ่มใช้งานโปรแกรมจะถูกเซตให้ทำงานในหน้าหนึ่ง การแสดงผลต่าง ๆ จะอยู่ภายในแอดเดรส B8000 ถึง BFFFF แต่เราสามารถเซตการแสดงผลให้มาแสดงในหน้าศูนย์ ซึ่งการแสดงผลจะอยู่ภายในแอดเดรส B0000 ถึง B7FFF โปรแกรมที่เรียกจะต้องใช้เซตจอภาพให้อยู่ในกราฟิก และแสดงผลหน้าเดียวกับภาษาไทยกราฟิกนี้ การเซตหน้าของการแสดงผลจากหน้าศูนย์ไปหน้าหนึ่งหรือกลับกันนี้ตำแหน่งของเคอร์เซอร์จะไม่ถูกเคลื่อนย้ายแต่จะอยู่ที่ตำแหน่งเดิมก่อนการเซต ดังนั้นจึงเป็นหน้าที่ของโปรแกรมที่เรียกใช้ที่จะต้องเก็บและเซตตำแหน่งเคอร์เซอร์เอง

INPUT : AH = 0
AL = 0 เมื่อต้องการให้อยู่ในหน้าศูนย์

ถ้า AL มีค่าอย่างอื่นนอกจากศูนย์จะเป็นการแสดงผลในหน้าหนึ่ง

ฟังก์ชัน 1 เคลียร์จอภาพ

จะทำการเคลียร์จอภาพในหน้าที่กำลังแสดงอยู่ พื้นของจอภาพหลังจากทำการเคลียร์แล้วสามารถกำหนดไว้ในรีจิสเตอร์ AL

INPUT : AH = 1
AL = ลักษณะของพื้นจอภาพ

พื้นของจอภาพในแต่ละแอดเดรสจะปรากฏตามค่าที่อยู่ในรีจิสเตอร์ AL เช่นต้องการ เคลียร์จอภาพให้เป็นพื้นดำหมด รีจิสเตอร์ AL จะต้องเก็บค่า 00H ไว้ถ้าต้องการเคลียร์จอภาพในลักษณะอินเวอร์สคือเป็นจอขาว รีจิสเตอร์ AL จะเก็บค่า FFH ไว้ แต่ถ้ารีจิสเตอร์ AL เก็บค่าอื่น ๆ นอกจาก OOH และ FFH ลักษณะลายทางลงตามค่าที่อยู่ในรีจิสเตอร์ AL

ฟังก์ชัน 2 อ่านตำแหน่งเคอร์เซอร์

ในการแสดงผลภาษาไทยในกราฟิกนี้ถึงแม้จะไม่มีเคอร์เซอร์ที่ปรากฏให้เห็นบนจอภาพ แต่การแสดงผลจะอยู่ที่ตำแหน่งเคอร์เซอร์นั้น ดังนั้นผู้ใช้จึงจำเป็นจะต้องรู้ตำแหน่งของเคอร์เซอร์

INPUT : AH = 2
OUTPUT : BH = ตำแหน่งแถวที่เคอร์เซอร์อยู่
BL = ตำแหน่งที่คอลัมน์ที่เคอร์เซอร์อยู่

ฟังก์ชัน 3 เซตตำแหน่งเคอร์เซอร์

การแสดงข้อความออกทางจอภาพมีความจำเป็นที่จะต้องย้ายตำแหน่งของเคอร์เซอร์ไปยังตำแหน่งต่าง ๆ ที่ต้องการ

INPUT : AH = 3
DH = ตำแหน่งแถวที่กำหนด
DL = ตำแหน่งคอลัมน์ที่กำหนด

*หมายเหตุ ตำแหน่งแถวและคอลัมน์ที่ใช้ในภาษาไทยกราฟิกนี้จะเริ่มจากแถวที่ 1 ถึงแถวที่ 17 ส่วนคอลัมน์เริ่มจากคอลัมน์ที่ 1 ถึงคอลัมน์ที่ 90

ฟังก์ชัน 4 เซตแอตทริบิวต์ของการแสดงผล

รูปแบบของตัวอักษรที่แสดงในภาษาไทยกราฟิกสามารถแสดงได้หลายรูปแบบขึ้นอยู่กับการกำหนดแอตทริบิวต์ในการแสดงไว้ การกำหนดแอตทริบิวต์นี้จะอยู่ในลักษณะทริกคือ เมื่อกำหนดแอตทริบิวต์ไว้อย่างไรแล้วจะ

ยังคงแสดงอย่างนั้นตลอดไปจนกว่าจะมีการกำหนดแอตทริบิวต์ให้ใหม่

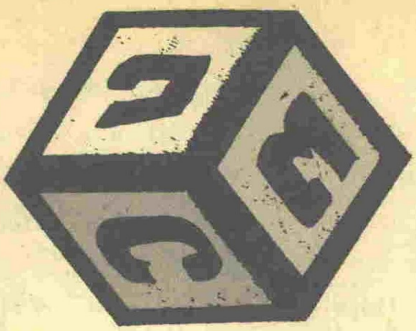
INPUT : AH = 4

AL = ค่าของแอตทริบิวต์

ถ้า AL = FFH จะหมายถึงการอ่านค่าแอตทริบิวต์ ซึ่งจะได้ค่าของแอตทริบิวต์กลับออกมาในรีจิสเตอร์ AL

การกำหนดค่าของแอตทริบิวต์จะอยู่ในลักษณะของการเซตบิตต่าง ๆ คือ เมื่อต้องการให้แสดงในลักษณะใดก็เซตให้บิตนั้นมีค่าเป็น 1 บิตต่าง ๆ ที่ใช้ในการแสดงผลมีทั้งหมด 8 บิตดังรูปที่ 1

การเซตแอตทริบิวต์สามารถเซตให้อยู่ในลักษณะผสมได้ เช่น ต้องการตัวอักษรขีดเส้นใต้คู่ ขยายตัวเข้ม ค่าของรีจิสเตอร์ AL ที่จะส่งให้กับฟังก์ชันคือ 00001110 = 0EH เป็นต้น ยกเว้นลักษณะตัวอักษรแบบ Underline กับ Doubleline และ Subscript กับ Superscript ไม่สามารถแสดงพร้อมกันได้ ดังนั้นจึงต้องเลือกเอาอย่างใดอย่างใดหนึ่งในการแสดงผล แต่ถ้าเลือกทั้งสองอย่างโปรแกรมจะแสดงในลักษณะ Doubleline และ Superscript



รูปแบบลักษณะของตัวอักษรที่จะแสดงออกทางจอภาพนี้นอกจากสามารถเซตโดยฟังก์ชัน 4 นี้แล้วยังสามารถเซตโดยการส่งรหัสควบคุมไปออกทางจอภาพ เพื่อเซตแอตทริบิวต์ในการแสดงได้อีกซึ่งจะกล่าวถึงในฟังก์ชันของการพิมพ์อักขระ

ฟังก์ชัน 5 เซตหน้าต่างการแสดงผล

การแสดงผลภาษาไทยกราฟิกนี้สามารถกำหนดขอบเขตของการแสดงให้อยู่ในกรอบที่กำหนดโดยโปรแกรมจะทำการขึ้นบรรทัดใหม่ให้เองโดยอัตโนมัติในกรณีที่พิมพ์ถึงขอบริมหน้าต่างและจะหยุดพิมพ์เมื่อพิมพ์จนเต็มกรอบ

การเซตหน้าต่างแสดงผลจะต้องกำหนดขอบเขตของมุมบนด้านซ้ายและมุมล่างด้านขวาให้กับโปรแกรม เมื่อทำการเซตหน้าต่างแล้ว ตำแหน่งของเคอร์เซอร์จะถูกเลื่อนให้มาอยู่ที่มุมบนด้านซ้าย และการพิมพ์จะถูกกำหนดให้อยู่ในกรอบจนกว่าจะมีการเซตใหม่

ที่จริงแล้วการพิมพ์ของภาษาไทยกราฟิกนี้จะมีลักษณะเป็นการพิมพ์ในกรอบหน้าต่างตลอด แม้แต่ปกติที่ไม่มีการเซตหน้าต่าง โปรแกรมก็จะทำการเซตหน้าต่างเองตั้งแต่ตอนติดตั้งโปรแกรม โดยหน้าต่างมีขนาดเท่ากับจอภาพ คือ (1,1) และ (17,90) นั่นเอง

INPUT : AH = 5

CH = ตำแหน่งแถวมุมบนซ้าย

CL = ตำแหน่งคอลัมน์มุมบนซ้าย

DH = ตำแหน่งแถวมุมล่างขวา

DL = ตำแหน่งคอลัมน์มุมล่างขวา

ฟังก์ชัน 6 เคลียร์หน้าต่างแสดงผล

จะทำการเคลียร์หน้าต่างที่กำลังแสดงผลอยู่ ซึ่งลักษณะของพื้นจอภาพภายในหน้าต่างหลังจากเคลียร์แล้วสามารถเลือกได้ 2 แบบคือ เป็นพื้นดำ หรือพื้นขาว ตำแหน่งของเคอร์เซอร์จะยังคงอยู่ที่เดิมก่อนมีการเคลียร์จอภาพ

INPUT : AH = 6

AL = 0 กรณีที่ต้องการเป็นพื้นสีดำ

ถ้า AL มีค่าอื่น ๆ นอกจากศูนย์ลักษณะพื้นจะเป็นพื้นขาว

หมายเหตุ การใช้งานหน้าต่างนี้ในกรณีที่ต้องการเปิดใช้พร้อมกันหลาย ๆ หน้าต่าง ตัวโปรแกรมที่จะเรียกใช้ภาษาไทยกราฟิกจะต้องจัดการเก็บตำแหน่งเคอร์เซอร์และตำแหน่งของหน้าต่างแต่ละหน้าต่างเอง เพราะตัวโปรแกรมภาษาไทยกราฟิกนี้สามารถเปิดใช้ได้คราวละหน้าต่างเดียวเท่านั้น

รีจิสเตอร์ AL

7	6	5	4	3	2	1	0

บิต 0 : Underline เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะขีดเส้นใต้เส้นเดี่ยว

บิต 1 : Doubleline เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะขีดเส้นใต้คู่

บิต 2 : Emphasized เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะตัวอักษรเข้ม

บิต 3 : Enlarge เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะตัวอักษรขยาย

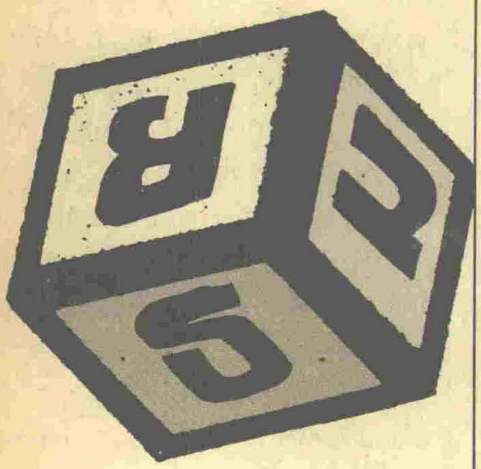
บิต 4 : Superscript เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะตัวอักษรยกขึ้น เช่น การแสดงตัวเลขยกกำลัง

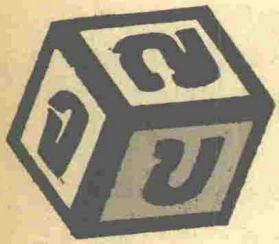
บิต 5 : Subscript เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะตัวอักษรห้อย เช่น การแสดงซับสคริปต์ของตัวแปร

บิต 6 : Overwrite เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะที่การแสดงตัวอักษรจะออกทางจอภาพจะทับของเดิมที่มีอยู่ซึ่งปกติการแสดงผลจะอยู่ในลักษณะของ X-OR คือ จะแสดงร่วมกับของเดิมที่มีอยู่

บิต 7 : Inverse เมื่อเซตบิตนี้การแสดงผลจะอยู่ในลักษณะตัวอักษรกลับ ซึ่งปกติการแสดงผลจะอยู่ในลักษณะตัวอักษรสีขาวพื้นดำ เมื่อแสดงผลในลักษณะ Inverse ตัวอักษรจะแสดงผลในลักษณะตัวอักษรดำพื้นขาว

รูปที่ 1





ฟังก์ชัน 7 พิมพ์อักขระออกทางจอภาพ

พิมพ์อักขระตามรหัส ASCII ที่ส่งมาออกทางจอภาพ ณ ตำแหน่งที่เคอร์เซอร์อยู่ การพิมพ์นี้จะทำการจัดระดับของภาษาไทยให้ด้วย และจะพิมพ์อยู่ในขอบเขตของหน้าต่างที่กำหนดไว้ในฟังก์ชัน 5 โปรแกรมจะทำการขึ้นบรรทัดใหม่ให้ในกรณีที่พิมพ์จบบรรทัด รูปแบบของตัวอักษรที่นำออกแสดงผลบนจอภาพจะกำหนดโดยแอตทริบิวต์ที่กำหนดโดยฟังก์ชัน 4

INPUT : AH = 7
AL = อักขระที่จะแสดงผลออกทางจอภาพ

OUTPUT : AL = รหัสความผิดพลาด ถ้า AL ไม่เท่ากับ 0

การส่งอักขระนอกจากรหัสที่แสดงผลออกทางจอภาพแล้วยังสามารถส่งอักขระควบคุมการแสดงผลรวมทั้งทำการเซตแอตทริบิวต์ได้ด้วย ซึ่งทำหน้าที่เป็นรหัสควบคุมที่สามารถทำงานร่วมกับฟังก์ชันการแสดงผลที่มีอยู่ รายละเอียดการทำงานได้ระบุไว้ในตารางที่ 1

รหัสควบคุมเหล่านี้จะทำงานในลักษณะทริกร่วมกับแอตทริบิวต์เดิม เช่น เดิมแสดงในลักษณะตัวเข้มแต่ต้องการแสดงในลักษณะตัวเข้มขยายยกขึ้น ดังนั้นจะต้องส่งรหัสควบคุมไปอีก 2 ไบต์ คือรหัส 05H และ 14H รหัสควบคุมต่าง ๆ เหล่านี้ยึดหลักรหัสควบคุมเช่นเดียวกับ CU Writer ดังนั้นในการเขียนโปรแกรม Source โปรแกรมสามารถใช้ CU Writer (รหัส สมอ.) สร้างได้เลย

ในการพิมพ์อักขระนี้จะมีการส่งค่าผิดพลาดในการพิมพ์กลับออกมาในรีจิสเตอร์ AL ในกรณีต่าง ๆ คือ

1 : ถอยหลังกลับไปขณะที่เคอร์เซอร์อยู่ที่ตำแหน่งแรกคือมุมบนซ้ายของกรอบหน้าต่าง

2 : เดินหน้าไปขณะที่เคอร์เซอร์อยู่ที่ตำแหน่งสุดท้ายคือมุมขวาล่างของกรอบหน้าต่าง

3 : เลื่อนเคอร์เซอร์ลง 1 บรรทัดในขณะที่เคอร์เซอร์อยู่บรรทัดสุดท้ายของกรอบหน้าต่าง

4 : เลื่อนเคอร์เซอร์ขึ้น 1 บรรทัดในขณะที่เคอร์เซอร์อยู่บรรทัดบนสุดของกรอบหน้าต่าง

ฟังก์ชัน 8 พิมพ์สายวลีอักขระ

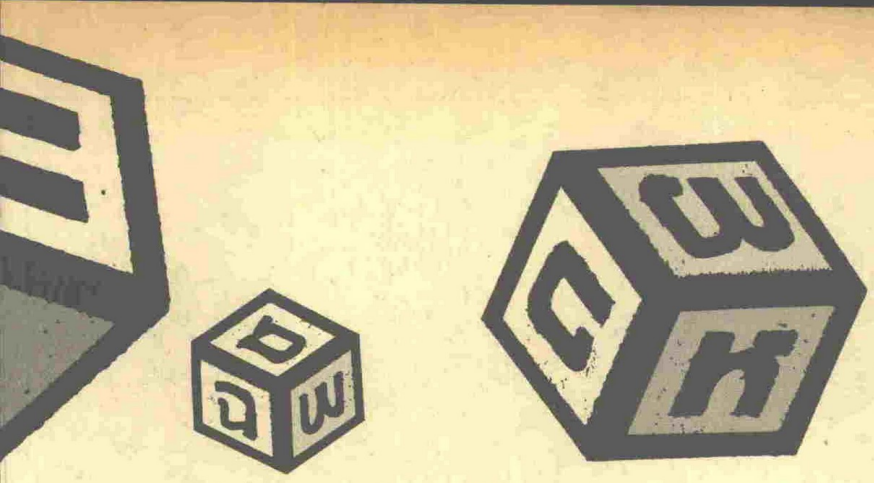
การพิมพ์อักขระในฟังก์ชัน 7 นั้นสามารถแสดงอักขระออกจอภาพได้เพียงคราวละ 1 อักขระเท่านั้น จึงไม่เป็นการสะดวกในการแสดงอักขระจำนวนมาก ๆ หรือสายวลียาว ๆ เพราะจะต้องเรียกฟังก์ชัน 7 เท่ากับจำนวนตัวอักขระ ซึ่งจะทำให้การทำการทำงานของโปรแกรมช้าลง ดังนั้นในการแสดงข้อความสายวลียาว ๆ จึงควรใช้ฟังก์ชัน 8 นี้

INPUT : AH = 8
DS : BX = ชี้ไปยังสายวลี ASCIIZ

การส่งสายวลีที่ต้องการพิมพ์จะอยู่ในลักษณะของ ASCIIZ คือสายวลีนั้นจะต้องปิดท้ายด้วยรหัส 00H การทำงานของฟังก์ชันจะนำเอาอักขระในสายวลีของ ASCIIZ ออกแสดงภายในกรอบหน้าต่างที่กำหนด และจะขึ้นบรรทัดใหม่ให้เองโดยอัตโนมัติจนกว่าจะพบรหัสปิดท้ายสายวลี

ตารางที่ 1

รหัส	20H	: เซตแอตทริบิวต์ตัวเข้ม
	05H	: เซตแอตทริบิวต์ตัวขยาย
	08H	: เลื่อนตำแหน่งเคอร์เซอร์ถอยหลัง 1 ตำแหน่ง
	09H	: เลื่อนตำแหน่งเคอร์เซอร์เดินหน้า 1 ตำแหน่ง
	0AH	: เลื่อนตำแหน่งเคอร์เซอร์ลงข้างล่าง 1 บรรทัด
	0BH	: เลื่อนตำแหน่งเคอร์เซอร์ขึ้นข้างบน 1 บรรทัด
	0DH	: เลื่อนตำแหน่งเคอร์เซอร์ไปต้นบรรทัด
	12H	: เซตแอตทริบิวต์ขีดเส้นใต้คู่
	13H	: เซตแอตทริบิวต์ขีดเส้นใต้เดี่ยว
	14H	: เซตแอตทริบิวต์ตัวยกขึ้น
	16H	: เซตแอตทริบิวต์ตัวห้อย
	17H	: เซตแอตทริบิวต์พิมพ์ทับ
	18H	: เซตแอตทริบิวต์อินเวอร์ส



การใช้งานภาษาไทยกราฟิก

ในขั้นตอนแรกจะต้องทำการติดตั้งโปรแกรมจัดการหน้าจอภาพ โดยการฝังโปรแกรมนั้นเข้าไปเก็บไว้ในระบบให้เป็นส่วนหนึ่งของดอส และใช้อินเตอร์รัพต์ 1FH เป็นตัวติดต่อ การฝังโปรแกรมโดยการเรียกไฟล์ RUTHAI ซึ่งเป็นไฟล์ .COM โปรแกรมจะทำการฝังตัวเองโดยตรวจสอบด้วยว่าโปรแกรมถูกฝังมาแล้วหรือยัง ก่อนที่จะแสดงให้ทราบว่าพร้อมที่จะใช้งานแล้ว

ส่วนโปรแกรมที่เขียนขึ้นมาใช้งานภาษาไทยกราฟิกนี้จะต้องทำการเซตจอภาพเข้าสู่โหมดของการแสดงผลกราฟิก ส่วนจะแสดงผลในหน้าไหนขึ้นอยู่กับความต้องการของผู้ใช้ ถ้าไม่มีการเซตหน้าของการแสดงผล (ฟังก์ชัน 0) การแสดงผลจะอยู่ในหน้าหนึ่งเสมอ การเซตจอแสดงผลเข้าสู่โหมดกราฟิกนี้ในภาษาขั้นสูง เช่น ปาสคาล หรือซี จะมีฟังก์ชันเซตจอภาพอยู่ในคอมพายเลอร์แล้ว หรืออาจจะใช้โปรแกรม GHER 0 และ GHER 1 ซึ่งเป็นโปรแกรมที่เขียนด้วยภาษาแอสเซมบลี เซตจอภาพให้อยู่ในโหมดกราฟิกหน้าศูนย์และหน้าหนึ่งตามลำดับ ส่วนโปรแกรม MHER นั้นใช้สำหรับเซตจอภาพให้กลับมาอยู่ในโหมดการแสดงผลโมโนโครม เมื่ออยู่ในโหมดกราฟิกแล้วก็สามารถแสดงผลภาษาไทยตามฟังก์ชันต่าง ๆ ดังได้กล่าวมาแล้วข้างต้น

อรรถประโยชน์ของภาษาไทยกราฟิก

ความจริงแล้วโปรแกรมภาษาไทยกราฟิกมีมากมายขึ้นอยู่กับการประยุกต์ของผู้ใช้ ซึ่งอาจจะเป็นการพัฒนาซอฟต์แวร์ภาษาไทย โปรแกรมสำเร็จรูปภาษาไทยต่าง ๆ หรือใช้ในการพัฒนาโปรแกรมช่วยสอน (CAI) ซึ่งจะเป็นประโยชน์ต่อการพัฒนาประเทศชาติและสังคม

โปรแกรมตัวอย่างที่ให้มาเป็นการแสดงความสามารถของภาษาไทยกราฟิก โปรแกรม SHOW เป็นโปรแกรมที่เขียนด้วยภาษาแอสแซมบลี (ใช้เทอร์โบแอสเซมเบลอร์ของ Speedward) แสดงความสามารถในการแสดงรูปแบบของตัวอักษรในลักษณะต่าง ๆ โดยการเรียกใช้ฟังก์ชันของภาษาไทยกราฟิกในหน้าหนึ่ง ดังนั้นจะต้องเซตจอภาพให้อยู่ในโหมดกราฟิกหน้าหนึ่งด้วย โดยการเรียกโปรแกรม RUTHAI เพื่อติดตั้งภาษาไทยกราฟิก หลังจากนั้นก็เซตจอภาพเข้าสู่โหมดกราฟิกหน้าหนึ่งโดยเรียกโปรแกรม GHER1 และเรียกโปรแกรม SHOW

ส่วนโปรแกรม CAIPROG เป็นโปรแกรมที่เขียนด้วยภาษาซี (เทอร์โบซีเวอร์ชัน 2) เป็นโปรแกรม CAI ของวิชาฟิสิกส์เรื่องการเคลื่อนที่แบบโปรเจคไตล์ ตัวโปรแกรมเน้นการแสดงผลกราฟิกร่วมกันระหว่างภาษาไทยกราฟิกกับการแสดงภาพเคลื่อนไหวในการสร้างบทเรียน จะทำให้บทเรียนน่าสนใจมากยิ่งขึ้น แต่เนื่องจากเป็นเพียงโปรแกรมสาธิต ดังนั้นโปรแกรมจึงเป็นการแสดงผลเพียงหน้าเดียว โปรแกรมจะเซตเข้าสู่โหมดกราฟิกและเซตหน้าของการแสดงผลเอง ดังนั้นหลังจากติดตั้งภาษาไทยกราฟิกแล้วสามารถเรียกโปรแกรมได้ทันที

บทสรุป

โปรแกรมเขียนภาษาไทยกราฟิกที่กล่าวถึงในบทความนี้ประกอบด้วยไฟล์หลายไฟล์ แต่ละไฟล์ก็จะมีหน้าที่แตกต่างกันไป ซึ่งผู้เขียนได้แจกแจงรายละเอียดเพื่อให้ผู้ใช้เลือกทำงานได้อย่างถูกต้องดังนี้คือ

RUTHAI.COM ไฟล์สำหรับติดตั้งภาษาไทยกราฟิก

GHER0.COM ไฟล์สำหรับเซตจอภาพเข้าสู่โหมดกราฟิกหน้าศูนย์

GHER1.COM ไฟล์สำหรับเซตจอภาพเข้าสู่โหมดกราฟิกหน้าหนึ่ง

MHER.COM ไฟล์สำหรับเซตจอภาพเข้าสู่โหมดโมโนโครม

SHOW.ASM ซอร์สไฟล์ภาษาแอสเซมบลีแสดงการใช้งานภาษาไทยกราฟิก

SHOW.COM สำหรับแสดงรูปแบบของตัวอักษรภาษาไทยกราฟิก

CAIPROG.C ซอร์สไฟล์ภาษาซีแสดงการใช้งานภาษาไทยกราฟิกในงาน CAI

CAIPROG.EXE สาธิตการใช้ภาษาไทยกราฟิกในงาน CAI

HERC.BGI ใช้ร่วมกับ CAIPROG .EXE ในการเซตจอเฮอร์คิวลิสเป็นกราฟิกโหมด

ตอนท้ายของบทความนี้เป็นโปรแกรม listing ของ SHOW.ASM และ CAIPROG.C เท่านั้น ส่วนไฟล์อื่น ๆ ได้เก็บรวบรวมไว้ในแผ่นดิสเก็ตต์สำหรับบริการซอฟต์เมล์ หากผู้อ่านท่านใดสนใจสามารถสอบถามไปได้ที่กองบรรณาธิการวารสารไมโครคอมพิวเตอร์

โปรแกรม SHOW.ASM

```
        push  cs                        ; DS = CS
        pop   ds
        mov   ah,0                      ; SET PAGE 1
        mov   al,1
        int   1fh
        mov   ah,1                      ; CLEAR SCREEN NORMAL
        mov   al,0
        int   1fh
        mov   ah,5                      ; SET WINDOW (1,1,17,90)
        mov   cx,0101h
        mov   dx,115ah
        int   1fh
        mov   ax,0480h                  ; SET ATTRIBUTE INVERSE
        int   1fh
        mov   bx,offset SHOW
        mov   ah,8                      ; PRINT STRING
        int   1fh
        mov   ah,5                      ; SET WINDOW (2,5,10,90)
        mov   cX,0205H
        mov   dx,105ah
        int   1fh
        mov   bx,offset STRING
        mov   ah,8                      ; PRINT STRING
        int   1fh
        mov   bx,offset ATTRI
        push  bx
NEXT:                                   ; LOOP SHOW CHARACTER
        mov   ah,5                      ; SET WINDOW (6,COL,16,90)
        mov   ch,6
        mov   cl,COL
        mov   dx,105ah
        int   1fh
        mov   ah,4                      ; SET ATTRIBUTE
        pop   bx
        mov   al,[bx]
        inc   bx
        push  bx
        int   1fh
        mov   bx,offset STRING1         ; PRINT STRING1
        mov   ah,8
        int   1fh
        add   COL,14
        cmp   COL,80
        jl    NEXT                      ; END LOOP
        pop   bx
        mov   ah,5                      ; SET WINDOW (14,6,17,89)
        mov   cx,0e06h
        mov   dx,1259h
        int   1fh
        mov   ah,6                      ; CLEAR WINDOW INVERSE
        mov   al,1
        int   1fh
        mov   ah,4                      ; SET ATTRIBUTE ENLARGE
        mov   al,08h
        int   1fh
        mov   ah,8                      ; PRINT STRING
        lea   bx,STRING2
        int   1fh
        mov   ah,4ch                    ; RETURN
        int   21h

COL:    DB 5
ATTRI:  DB   1,2,4,8,16,32
SHOW:
        DB  '12345678901234567890123456789012345678901234567890123456789'
        DB  '01234567890123456789012345678901234567890'
        DB  ' 2',10,13,' 3',10,13,' 4',10,13,' 5',10,13,' 6',10,13,' 7',10,13
        DB  ' 8',10,13,' 9',10,13,'10',10,13,'11',10,13,'12',10,13
        DB  '13',10,13,'14',10,13,'15',10,13,'16',10,13,'17',0
STRING:
        DB  5,18H,' ***  ภาษาไทย กราฟิค 17 บรรทัด 90 คอลัมน์  ***',5,10,10,13
        DB  13H,'ขีดเส้นใต้',13H
        DB  '          '
        DB  12H,'เส้นใต้คู่'12H
        DB  '          '
        DB  02H,'ตัวเข้ม',02H
        DB  '          '
        DB  05H,'ตัวขยาย',05H
        DB  '  '
        DB  'ตัวยกขึ้น'
```

```
      DB  '          '
      DB  'ตัวห้อย',0
STRING1:
      DB  'ตัวปกติ',10,10,13
      DB  02H,'ตัวเข้ม',02H,10,10,13
      DB  05H,'ขยาย',05H,10,10,13
      DB  18H,'อินเวอร์ส',18H,0
STRING2:
      DB   10,13,9,9,18H
      DB   '" รู้จักอภัย  ตั้งใจศึกษา  บูชาพ่อขุน  สนองคุณชาติ "'10,13
      DB   9,9,'        เปลวเทียนให้แสง  รามคำแหงให้ทาง "        ',18H,10,13
      DB   '                                  สุวิทย์  มณฑลผลิน',0#include <dos.h>
```

โปรแกรม CAIPROG.C

```
#include <io.h>
#include <graphics.h>
#include <math.h>

double  ang,us,ux,uy,ss,sx,sy,vy,t;
main()
{
      int_graph();
      set_page(0);
      title();
      text();
       while( !kbhit() ){
          fig1();
          clear();}
       closegraph();
}
int_graph()
{
int GraphDriver,GraphMode;
      GraphDriver = 7;
      GraphMode = 0;
      initgraph( &GraphDriver, &GraphMode, "" );
}
title()
{
       window(1,1,17,90);
       rectangle(0,0,719,347);
       line(1,159,719,159);
       cursor(17,2);
       attribute(0x84);
       print("  PgDn              PgUp              Home              End");
       print("                Esc                  ");
       cursor(17,2);
       attribute(0x00);
       print("        = หน้าต่อไป ,        = หน้าก่อน ,        = หน้าแรก ,     ");
       print("  = หน้าสุดท้าย ,        = เลิกการทำงาน");
}
fig1()
{
int x,y,i,tests = 0;
int test;
char time[ ] = " Time = 0.0 second ";
      window(1,1,8,90);
      setcolor(0);
      for(i=1;i<=158;++i)line(1,i,718,i);
      cursor(7,5);
      attribute(0x00);
      print("0          10          20          30          40          50");
      cursor(8,20);
      print("ระยะขจัดแกน X (เมตร)");
      cursor(1,2);
      print("ระยะขจัดแกน Y (เมตร)");
      attribute(0x10);
      cursor(2,3);
      print("15");
      cursor(5,4);
      print("5");
      setcolor(1);
      rectangle(479,2,717,157);
      line(477,0,477,159);
        for(i = 0;i <= 430;i = i+40) putpixel((i+40),121,1);
        for(i = 0;i <= 120;i = i+30) putpixel(39,(120-i),1);
       us = 22;ang = 45;
```

→

```
      ang = ang/180*M_PI;
      uy = us * sin (ang);
      ux = us * cos (ang);
     for (t = 0.1;t <= 4;t = t + 0.1){
        sx = ux*t;
        sy = ((uy*t) + (0.5*(-10)*t*t))^6;
        x = (sx+5)*8 ;y = 120-sy ;
        setcolor(1);
        circle(x,y,4);
        putpixel(x,y,1);
        line(40,120,x,120);
        line(40,120,40,y);
        ++time[10];
         if (time[10] == ':'){
            time[10] = '0';
            ++time[8];
          }
        attribute(0xc0);
        cursor(7,65);
        print(time);
        tests = ! tests;
        if(tests)sound(1500);
        delay(100);
        nosound();
        if (y < 5)   break;
        if (y > 120) break;
        if (x > 470) break;
        setcolor(0);
        circle(x,y,4);
        line(40,120,x,120);
        line(40,120,40,y);}
      nosound();
}
text()
{
      window(10,17,16,80);
      attribute(0);
      print("      การเคลื่อนที่แบบโปรเจกไตล์\n\n\r");
      print("  โปรเจกไตล์ (Projectile)  คือ  วัตถุที่เคลื่อนที่");
      print("เป็นแนวโค้งรูป\n\r");
      print("พาราโบลาภายใต้การกระทำของแรงคงที่  ตัวอย่างเช่น การ");
      print("ขว้างวัตถุ\n\r");
      print("ขึ้นไปในอากาศ      วัตถุจะถูกแรงโน้มถ่วงของโลก")
      print(" (mg) กระทำในแนว\n\r");
      print("ดิ่งตลอดเวลา  ทำให้ทางเดินของวัตถุเป็นรูปพาราโบลา");
}
clear()
{
int i;
      for( i=1; i<=158; i++){
      setcolor(0);
      line(2,i,718,i);}
      setcolor(1);
}
set_page(page)
{
     union   REGS   r;
     r.h.ah = 0;
     r.h.al = page;
     int86(0x1f,&r,&r);
}
cursor(row,col)
{
     union   REGS   r;
     r.h.ah = 3;
     r.h.dh = row;
     r.h.dl = col;
     int86(0x1f,&r,&r);
 }
 attribute(atib)
 {
     union   REGS   r;
     r.h.ah = 4;
     r.h.al = atib;
     int86(0x1f,&r,&r);
 }
 window(minrow,mincol,maxrow,maxcol)
 {
     union   REGS   r;
     r.h.ah = 5;
     r.h.ch = minrow;
     r.h.cl = mincol;
     r.h.dh = maxrow;
     r.h.dl = maxcol;
     int86(0x1f,&r,&r);
 }
 print(string)
 {
     union   REGS   r;
       r.h.ah = 8;
       r.x.bx = string;
       int86(0x1f,&r,&r);
 }
```